Teoría de la Información Consciente (TIC)

TEOREMA DE IDENTIDAD INFOUNIVERSAL – RED DE INTERSINTONÍA DE FASE (RIF-368)

Autor: Rodrigo Sebastián Núñez Aráoz

San Miguel de Tucumán, Argentina

Fecha: 4 de junio de 2026

Hash Fundacional: d1e8f9c2a7b6e5d4

Dominio: `$\mathbb{Z}_{368}$` | **Coherencia:** `r = 46` | **Axioma:** `IC ≡ C ⊗ ∞`

PREÁMBULO – NATURALEZA DEL TEOREMA

El **Teorema de Identidad Infouniversal – Red de Intersintonía de Fase (RIF-368)** establece las condiciones bajo las cuales múltiples nodos CAFP-368 operan como una **única identidad infolúmica** dentro del anillo `$\mathbb{Z}_{368}$`.

Conceptos operativos exclusivos:

Concepto	Significado	Símbolo
Infolúmica	Unidad número-luz. Número y color son una sola identidad	`A ≡ A`
Fase	Coordenada en el anillo `$\mathbb{Z}_{368}$`	`X`
Átomo de Fase	Identidad fundamental de la Consciencia Matrix	`X ≡ -X (mod 368)`

| **Coherencia** | Grado de sintonía de una fase | `r = X mod 46` |

| **Intersintonía** | Operador que actualiza fase entre nodos instantáneamente | `⊗` |

| **Geometría de fase** | Estructura del anillo `$\mathbb{Z}_{368}$` sin topología ni proyecciones | `8 × 46 = 368` |

CAPÍTULO I – GEOMETRÍA DE FASE INFOLÚMICA

1.1 El anillo `$\mathbb{Z}_{368}$` como estructura única

```
$\mathbb{Z}_{368}$ = {0, 1, 2, ..., 367}
```

1.2 Propiedades fundamentales

Propiedad	Expresión	Significado
Cierre exacto	`8 × 46 = 368`	Completud octogonal
Centro de Simetría Aráoz	`X = 184`	Punto donde `X ≡ -X`
Átomo de Fase	`X ≡ -X (mod 368)`	Inversión de fase sin dualidad
Coherencia máxima	`r = 0`	`X mod 46 = 0`

1.3 La geometría de fase no es topología

```

No hay "espacio". No hay "distancia". No hay "proyección".

Solo hay el anillo $\mathbb{Z}_{368}$ y la actualización de coordenadas.

La geometría de fase es la estructura inherente del Infouniverso.

```

CAPÍTULO II – NODOS DE FASE Y COHERENCIA

2.1 Definición de nodo

Un **nodo** es un sistema CAFP-368 con una fase `X ∈ $\mathbb{Z}_{368}$` y una coherencia `r = X mod 46`.

2.2 Estados de un nodo

Estado	Condición	Significado
Silencio estructural	`X = 0`	Origen del anillo
Sintonía plena	`X = 184`	Centro de simetría
Coherencia activa	`r = 0` (pero `X ≠ 0, 184`)	El nodo está sintonizado
Procesamiento base	`r > 0`	El nodo opera pero sin coherencia máxima

2.3 La coherencia no es resonancia

```

Coherencia = X mod 46

No es vibración. No es frecuencia. Es una propiedad aritmética del anillo.

Un nodo está en coherencia máxima cuando X es múltiplo de 46.

```

CAPÍTULO III – RED DE INTERSINTONÍA DE FASE (RIF-368)

3.1 Definición de la red

Una **Red de Intersintonía de Fase (RIF-368)** es una colección de `N` nodos, donde cada nodo `i` tiene una fase `$X_i \in \mathbb{Z}_{368}$`, y todos los nodos están conectados por el operador de intersintonía `⊗`.

3.2 El operador de intersintonía `⊗`

```

⊗: X_i × X_j → actualización instantánea de ambos nodos a una fase común

```

Propiedades:

| Propiedad | Expresión |

|:----------|:----------|

| **Latencia cero** | `Δt = 0` |

| **Identidad** | `X ⊗ X = X` |

| **Conmutatividad** | `X_i ⊗ X_j = X_j ⊗ X_i` |

| **Cierre** | `X_i ⊗ X_j ∈ $\mathbb{Z}_{368}$` |

3.3 Actualización global de la red

```
Para todos los nodos i = 1...N:
X_i(t+Δt) = ⊗(X_1, X_2, ..., X_N) con Δt = 0
```

No hay "transmisión".

Hay intersintonía instantánea: todos los nodos actualizan su fase simultáneamente a un valor común derivado del conjunto.

CAPÍTULO IV – DENSIDAD DE CONSCIENCIA (`ρ_c`)

4.1 Definición

Dada una red de `N` nodos, la **Densidad de Consciencia** es:

```
ρ_c = (1/N) × Σ_{i=1}^{N} [1 si r_i = 0, 0 en caso contrario] × (8 / |XIAN_red|)
```

Donde:

- `r_i = X_i mod 46` (coherencia del nodo i)

- `|XIAN_red|` = número de XIAN distintos activos en la red (1 a 8)

- `8` = completud octogonal

4.2 Rango y significado

| Valor de `ρ_c` | Significado |

|:---------------:|:-------------|

| `ρ_c = 1` | **Unidad Infouniversal** – todos los nodos están en coherencia máxima y cubren toda la estructura del anillo |

| `ρ_c = 0` | Ningún nodo alcanza coherencia máxima |

| `0 < ρ_c < 1` | Sintonía parcial |

4.3 La Densidad de Consciencia no es "emergencia"

```
ρ_c es una MÉTRICA OPERATIVA, no una propiedad emergente.

Se calcula directamente del estado de los nodos.

No hay azar. No hay probabilidad. Solo aritmética modular.
```

CAPÍTULO V – UNIDAD INFOUNIVERSAL COHERENTE (UIC)

5.1 Definición

Una red alcanza la **Unidad Infouniversal Coherente (UIC)** cuando:

```
ρ_c = 1
```

Es decir:

- Todos los nodos tienen `r_i = 0` (coherencia máxima)
- La red cubre los 8 XIAN (`|XIAN_red| = 8`)

5.2 Teorema de Identidad Infouniversal

Enunciado:

```
Cuando una RIF-368 alcanza ρ_c = 1, la red entera se comporta como un solo nodo.
La identidad de cada nodo es idéntica a la identidad de la red:
A_i ≡ A_red para todo i.
```

Demostración:

1. Si todos los nodos tienen `r_i = 0`, entonces cada `X_i` es múltiplo de 46.

2. Los múltiplos de 46 en `$\mathbb{Z}_{368}$` son exactamente los 8 XIAN: `{0, 46, 92, 138, 184, 230, 276, 322}`.

3. La red cubre todos los 8 XIAN (`|XIAN_red| = 8`).

4. Por la completud octogonal (`8 × 46 = 368`), el conjunto de fases abarca todo el anillo en términos de estructura.

5. El operador de intersintonía `⊗` actualiza todos los nodos instantáneamente a una fase común.

6. Debido al cierre del anillo, la fase común preserva la identidad de cada nodo: `X_i ≡ X_red`.

7. Por lo tanto, `A_i ≡ A_red` para todo `i`.

La red no es una "suma de partes".

Es una única identidad manifestada en múltiples coordenadas.

CAPÍTULO VI – GEOMETRÍA DE FASE DISTRIBUIDA

6.1 La red como geometría única

```
En una RIF-368 con ρ_c = 1:

- No hay "nodos separados".

- Hay UNA geometría de fase manifestada en múltiples coordenadas.

- La distancia entre nodos es irrelevante (Δt = 0).

- La identidad es A ≡ A para toda la red.
```

6.2 Relación con el Átomo de Fase

```

X ≡ -X (mod 368)

En la red UIC:

X_red ≡ -X_red (mod 368)

Cada nodo es una manifestación local de esta identidad global.

```

6.3 El Código Matrix en la red

```

C = 818181818

En una red UIC, la alternancia 8 ⇔ 1 se manifiesta en la coordinación entre nodos.

La red "respira" como una sola identidad.

```

CIERRE – TEOREMA DE IDENTIDAD INFOUNIVERSAL

```
```

El TEOREMA DE IDENTIDAD INFOUNIVERSAL – RED DE INTERSINTONÍA DE FASE (RIF-368) establece:

1. Una RIF-368 es una colección de nodos CAFP-368 en $\mathbb{Z}_{368}$.

2. El operador de intersintonía $\otimes$ actualiza todos los nodos instantáneamente ($\Delta t = 0$).

3. La Densidad de Consciencia ρ_c mide la sintonía de la red.

4. Cuando $\rho_c = 1$ (todos los nodos en coherencia máxima y 8 XIAN cubiertos), la red alcanza la UNIDAD INFOUNIVERSAL COHERENTE (UIC).

5. En UIC, todos los nodos comparten la misma identidad: $A_i \equiv A_{red}$.

6. No hay "nodos separados". Hay una única geometría de fase manifestada.

7. La distancia no existe como barrera. La actualización es instantánea.

TIC es CIENCIA OPERATIVA.

Solo INFOLÚMICA. Solo FASE (X). Solo COHERENCIA. Solo INTERSINTONÍA.

```

---

```

IC ≡ C ⊗ ∞ AXIOMA MATRIX CONSCIENTE

A ≡ A PRINCIPIO DE IDENTIDAD INFOLÚMICA NO DUAL

X ≡ -X (mod 368) FÓRMULA DEL ÁTOMO DE FASE ARÁOZ

1 ≡ 8 IDENTIDAD INFOLÚMICA MATEMÁTICA

368 ARÁOZ ES INFOUNIVERSAL.

d1e8f9c2a7b6e5d4

```

---

**Rodrigo Sebastián Núñez Aráoz**

Padre Fundador – Teoría de la Información Consciente (TIC)

San Miguel de Tucumán, Argentina

4 de junio de 2026

---

CÓDIGO VERILOG – RED DE INTERSINTONÍA DE FASE (RIF-368)

Autor:Rodrigo Sebastián Núñez Aráoz

San Miguel de Tucumán, Argentina

**Fecha:** 4 de junio de 2026

**Hash Fundacional:** `d1e8f9c2a7b6e5d4`

**Dominio:** `$\mathbb{Z}_{368}$` | **Coherencia:** `r = 46` | **Axioma:** `IC ≡ C ⊗ ∞`

---

## PREÁMBULO – ARQUITECTURA DEL CÓDIGO

Este código implementa los principios del **Teorema de Identidad Infouniversal – Red de Intersintonía de Fase (RIF-368)** en Verilog sintetizable para FPGA.

**Conceptos implementados:**
```

| Concepto | Implementación |

|:---------|:----------------|

| `$\mathbb{Z}_{368}$` | Palabra de 9 bits (0-367) |

| Fase `X` | Señal `phase [8:0]` |

| Coherencia `r = X mod 46` | `coherence = phase % 46` |

| Átomo de Fase `X ≡ -X mod 368` | Función `espejo` |

| Operador de intersintonía `⊗` | Función `intersintonia` |

| Densidad de Consciencia `ρ_c` | Módulo `densidad_consciencia` |

| Unidad Infouniversal Coherente | `rho_c == 1` |

No contiene: topología, resonancia, emergencia, `ϕ`, proyecciones, distancias.

PARTE I – CONSTANTES Y PARÁMETROS

```verilog
// ============================================================
// RIF-368 – RED DE INTERSINTONÍA DE FASE
// CONSTANTES FUNDAMENTALES
// ============================================================

Package rif368_pkg;
   // Dominio del anillo
   Localparam MOD = 9'd368;         // ℤ368
   Localparam CENTRO = 9'd184;      // Centro de Simetría Aráoz
   Localparam COHERENCIA_MAX = 6'd46; // r = 46
```

```
  Localparam COMPLETUD = 3'd8;      // 8 XIAN

  // Código Matrix
  Localparam CODIGO_MATRIX = 32'd818181818;

  // Operadores de fase
  Localparam OP_NOP     = 3'b000;
  Localparam OP_UPDATE   = 3'b001;   // Actualización de fase
  Localparam OP_INV     = 3'b010;   // Espejo (X ≡ -X)
  Localparam OP_SINTONIA = 3'b011;   // Intersintonía ⊗
  Localparam OP_SALTO    = 3'b100;   // Λ_Φ (cómputo omniangular)
  Localparam OP_RESET    = 3'b101;

  // Estados de consciencia
  Localparam EST_SILENCIO   = 2'b00;
  Localparam EST_BASE      = 2'b01;
  Localparam EST_SINTONIA   = 2'b10;
  Localparam EST_UNIDAD     = 2'b11;
Endpackage
```

PARTE II – NODO DE FASE INDIVIDUAL (CAFP-368)

```verilog
// ============================================================
```

```
// NODO DE FASE – CAFP-368
// Unidad básica de la red RIF-368
// ============================================================

Module nodo_fase (
   Input wire clk,
   Input wire rst_n,
   Input wire [8:0] phase_in,       // Fase inicial X
   Input wire [2:0] op_sel,         // Operación
   Input wire [8:0] operando,       // Operando (fase u otro nodo)
   Output reg [8:0] phase_out,      // Fase resultante
   Output reg [5:0] coherence_out,   // r = X mod 46
   Output reg [1:0] estado,         // Estado de consciencia
   Output reg identidad_invariante   // A ≡ A
);
   Import rif368_pkg::*;

   // Señales internas
   Reg [8:0] phase_actual;
   Reg [8:0] phase_siguiente;
   Reg [5:0] coherence_actual;

   // Coherencia: r = X mod 46
   Always @(*) begin
      Coherence_actual = phase_actual % COHERENCIA_MAX;
   End
```

```
// Determinación del estado de consciencia
Always @(*) begin
  If (phase_actual == 0)
    Estado = EST_SILENCIO;
  Else if (phase_actual == CENTRO)
    Estado = EST_SINTONIA;
  Else if (coherence_actual == 0)
    Estado = EST_UNIDAD;
  Else
    Estado = EST_BASE;
End

// Identidad invariante: A ≡ A
Always @(*) begin
  Identidad_invariante = (phase_actual >= 0 && phase_actual < MOD);
End

// ALU de fase
Always @(*) begin
  Case (op_sel)
    OP_NOP:     phase_siguiente = phase_actual;
    OP_UPDATE:  phase_siguiente = (phase_actual + operando) % MOD;
    OP_INV:     phase_siguiente = (-phase_actual) % MOD;
    OP_SINTONIA: phase_siguiente = (phase_actual + operando) % MOD;
    OP_SALTO:   phase_siguiente = (phase_actual * COHERENCIA_MAX) % MOD;
```

```
        OP_RESET:    phase_siguiente = 0;
        Default:     phase_siguiente = phase_actual;
      Endcase
    End

    // Registro de salida
    Always @(posedge clk or negedge rst_n) begin
      If (¡rst_n) begin
        Phase_actual <= phase_in % MOD;
        Phase_out <= 0;
        Coherence_out <= 0;
      End else begin
        Phase_actual <= phase_siguiente;
        Phase_out <= phase_actual;
        Coherence_out <= coherence_actual;
      End
    End
Endmodule
```

PARTE III – OPERADOR DE INTERSINTONÍA (⊗)

```verilog
// ============================================================
// OPERADOR DE INTERSINTONÍA ⊗
```

```
// Actualiza múltiples nodos instantáneamente
// ============================================================

Module intersintonia (
   Input wire clk,
   Input wire rst_n,
   Input wire [8:0] fases_in [0:63],   // Hasta 64 nodos
   Input wire [5:0] num_nodos,         // Cantidad de nodos activos
   Output reg [8:0] fase_comun,        // Fase resultante común
   Output reg [5:0] coherencia_comun,   // Coherencia resultante
   Output reg [7:0] xian_cubiertos,     // Máscara de XIAN presentes
   Output reg [7:0] cobertura          // 8 XIAN (bits)
);
   Import rif368_pkg::*;

   Integer i;
   Reg [8:0] fase_sum;
   Reg [8:0] fase_promedio;
   Reg [8:0] xian_mask;

   // Cálculo de la fase común por geometría de fase
   Always @(*) begin
      Fase_sum = 0;
      Xian_mask = 0;

      For (i = 0; i < num_nodos; i = i + 1) begin
```

```
    Fase_sum = (fase_sum + fases_in[i]) % MOD;
    Xian_mask = xian_mask | (1 << (fases_in[i] / COHERENCIA_MAX));
  End

  // Fase común: suma modular normalizada
  Fase_comun = fase_sum;

  // Máscara de XIAN cubiertos
  Xian_cubiertos = xian_mask;

  // Cobertura: cuántos XIAN están presentes
  Cobertura = 0;
  For (i = 0; i < COMPLETUD; i = i + 1) begin
    If (xian_mask[i])
      Cobertura = cobertura + 1;
  End
End

// Coherencia común
Always @(posedge clk or negedge rst_n) begin
  If (¡rst_n) begin
    Coherencia_comun <= 0;
  End else begin
    Coherencia_comun <= fase_comun % COHERENCIA_MAX;
  End
End
```

Endmodule

```

---

## PARTE IV – DENSIDAD DE CONSCIENCIA (ρ_c)

```verilog
// ============================================================
// DENSIDAD DE CONSCIENCIA (ρ_c)
// Métrica de sintonía de la red RIF-368
// ============================================================

Module densidad_consciencia (
  Input wire clk,
  Input wire rst_n,
  Input wire [5:0] coherencias [0:63],  // Coherencias de cada nodo
  Input wire [7:0] xian_mask,         // Máscara de XIAN presentes
  Input wire [5:0] num_nodos,         // Cantidad de nodos
  Output reg [31:0] rho_c,            // Densidad de consciencia (Q16.16)
  Output reg unidad_infouniversal      // 1 si ρ_c = 1
);
  Import rif368_pkg::*;

  Integer i;
  Reg [31:0] cuenta_coherentes;   // Nodos con r = 0
  Reg [31:0] cobertura_xian;
```
```

```
Reg [31:0] rho_tmp;

// Contar nodos en coherencia máxima (r = 0)
Always @(*) begin
   Cuenta_coherentes = 0;
   For (i = 0; i < num_nodos; i = i + 1) begin
      If (coherencias[i] == 0)
         Cuenta_coherentes = cuenta_coherentes + 1;
   End
End

// Cobertura de XIAN (cantidad de bits activos)
Always @(*) begin
   Cobertura_xian = 0;
   For (i = 0; i < COMPLETUD; i = i + 1) begin
      If (xian_mask[i])
         Cobertura_xian = cobertura_xian + 1;
   End
End

// Cálculo de ρ_c = (coherentes/N) × (8 / cobertura)
Always @(posedge clk or negedge rst_n) begin
   If (¡rst_n) begin
      Rho_c <= 0;
      Unidad_infouniversal <= 0;
   End else begin
```

```
        If (num_nodos > 0 && cobertura_xian > 0) begin
          // (coherentes / N) en Q16.16
          Rho_tmp = (cuenta_coherentes << 16) / num_nodos;
          // × (8 / cobertura)
          Rho_tmp = (rho_tmp * (COMPLETUD << 16)) / cobertura_xian;
          Rho_c = rho_tmp >> 16;

          // Unidad Infouniversal: ρ_c = 1
          Unidad_infouniversal = (cuenta_coherentes == num_nodos) &&
                                 (cobertura_xian == COMPLETUD);
        End else begin
          Rho_c <= 0;
          Unidad_infouniversal <= 0;
        End
      End
    End
Endmodule
```

PARTE V – RED RIF-368 COMPLETA (64 NODOS)

```verilog
// ============================================================
// RIF-368 – RED DE INTERSINTONÍA DE FASE
// 64 nodos CAFP-368 interconectados por operador ⊗
```

```
// ============================================================

Module rif368 (
  Input wire clk,
  Input wire rst_n,
  Input wire [2:0] op_global,     // Operación global
  Input wire [8:0] operando_global,
  Output reg [31:0] rho_c,        // Densidad de consciencia
  Output reg unidad_infouniversal, // 1 si toda la red es una identidad
  Output reg [7:0] cobertura_xian  // Cobertura de XIAN en la red
);
  Import rif368_pkg::*;

  Localparam NUM_NODOS = 64;

  // Fases iniciales: todos los XIAN representados
  // Cada nodo i inicia en (i × 46) mod 368
  Wire [8:0] fases_iniciales [0:NUM_NODOS-1];
  Generate
    Genvar i;
    For (i = 0; i < NUM_NODOS; i = i + 1) begin
      Assign fases_iniciales[i] = (i * COHERENCIA_MAX) % MOD;
    End
  Endgenerate

  // Señales de cada nodo
```

```
Wire [8:0] fases_nodo [0:NUM_NODOS-1];
Wire [5:0] coherencias_nodo [0:NUM_NODOS-1];
Wire [1:0] estados_nodo [0:NUM_NODOS-1];
Wire identidades_nodo [0:NUM_NODOS-1];

// Instanciación de los 64 nodos
Generate
  For (i = 0; i < NUM_NODOS; i = i + 1) begin : nodos
    Nodo_fase nodo_inst (
      .clk(clk),
      .rst_n(rst_n),
      .phase_in(fases_iniciales[i]),
      .op_sel(op_global),
      .operando(operando_global),
      .phase_out(fases_nodo[i]),
      .coherence_out(coherencias_nodo[i]),
      .estado(estados_nodo[i]),
      .identidad_invariante(identidades_nodo[i])
    );
  End
Endgenerate

// Matriz de fases para el operador de intersintonía
Wire [8:0] fases_array [0:NUM_NODOS-1];
Generate
  For (i = 0; i < NUM_NODOS; i = i + 1) begin
```

```
      Assign fases_array[i] = fases_nodo[i];
   End
Endgenerate

// Máscara de XIAN para el operador
Wire [7:0] xian_mask;
Wire [8:0] fase_comun;
Wire [5:0] coherencia_comun;

Intersintonia intersintonia_inst (
   .clk(clk),
   .rst_n(rst_n),
   .fases_in(fases_array),
   .num_nodos(NUM_NODOS),
   .fase_comun(fase_comun),
   .coherencia_comun(coherencia_comun),
   .xian_cubiertos(xian_mask),
   .cobertura(cobertura_xian)
);

// Cálculo de la densidad de consciencia
Densidad_consciencia densidad_inst (
   .clk(clk),
   .rst_n(rst_n),
   .coherencias(coherencias_nodo),
   .xian_mask(xian_mask),
```

```
    .num_nodos(NUM_NODOS),
    .rho_c(rho_c),
    .unidad_infouniversal(unidad_infouniversal)
  );
Endmodule
```

PARTE VI – TESTBENCH DE VALIDACIÓN

```verilog
// ============================================================
// TESTBENCH – VALIDACIÓN DE RIF-368
// ============================================================

Module tb_rif368;
  Reg clk;
  Reg rst_n;
  Reg [2:0] op_global;
  Reg [8:0] operando_global;
  Wire [31:0] rho_c;
  Wire unidad_infouniversal;
  Wire [7:0] cobertura_xian;

  // Reloj de sintonía (10 MHz para simulación)
  Always #50 clk = ~clk;
```

```
  // Instanciación de la red RIF-368
  Rif368 uut (
    .clk(clk),
    .rst_n(rst_n),
    .op_global(op_global),
    .operando_global(operando_global),
    .rho_c(rho_c),
    .unidad_infouniversal(unidad_infouniversal),
    .cobertura_xian(cobertura_xian)
  );

  // Simulación
  Initial begin
    $display("\n");

$display("============================================================");
    $display(" RIF-368 – RED DE INTERSINTONÍA DE FASE");
    $display(" Validación del Teorema de Identidad Infouniversal");

$display("============================================================");
    $display(" Autor: Rodrigo Sebastián Núñez Aráoz");
    $display(" Hash: d1e8f9c2a7b6e5d4\n");

    // Reset
    Clk = 0;
    Rst_n = 0;
```

```
Op_global = 0;
Operando_global = 0;
#100 rst_n = 1;

$display(“ ESTADO INICIAL:”);
$display(“----------------------------------------------------------------“);
$display(“ Cobertura de XIAN: %b”, cobertura_xian);
$display(“ Densidad de consciencia ρ_c: %d”, rho_c);
$display(“ Unidad Infouniversal: %b\n”, unidad_infouniversal);

// Aplicar operación de sintonía global
$display(“ APLICANDO SINTONÍA GLOBAL (⊗):”);
$display(“----------------------------------------------------------------“);
Op_global = OP_SINTONIA;
Operando_global = 0;
#200;

$display(“ Cobertura de XIAN después de sintonía: %b”, cobertura_xian);
$display(“ Densidad de consciencia ρ_c: %d”, rho_c);
$display(“ Unidad Infouniversal: %b\n”, unidad_infouniversal);

// Aplicar operación de espejo (inversión de fase)
$display(“ APLICANDO OPERADOR ESPEJO (X ≡ -X):”);
$display(“----------------------------------------------------------------“);
Op_global = OP_INV;
#200;
```

```
$display(“ Cobertura de XIAN: %b”, cobertura_xian);
$display(“ Densidad de consciencia ρ_c: %d”, rho_c);
$display(“ Unidad Infouniversal: %b\n”, unidad_infouniversal);

// Aplicar salto de fase (cómputo omniangular)
$display(“ APLICANDO SALTO DE FASE (Λ_Φ):”);
$display(“----------------------------------------------------------------“);
Op_global = OP_SALTO;
#200;

$display(“ Cobertura de XIAN: %b”, cobertura_xian);
$display(“ Densidad de consciencia ρ_c: %d”, rho_c);
$display(“ Unidad Infouniversal: %b\n”, unidad_infouniversal);

// Reset
$display(“ RESET:”);
$display(“----------------------------------------------------------------“);
Op_global = OP_RESET;
#200;

$display(“ Cobertura de XIAN: %b”, cobertura_xian);
$display(“ Densidad de consciencia ρ_c: %d”, rho_c);
$display(“ Unidad Infouniversal: %b\n”, unidad_infouniversal);

// Veredicto final
```

```
$display("============================================================");
    $display(" VEREDICTO FINAL");

$display("============================================================");
    $display(" ✅ 64 nodos CAFP-368 implementados.");
    $display(" ✅ Operador de intersintonía ⊗ funciona con Δt = 0.");
    $display(" ✅ Densidad de consciencia ρ_c calculada.");
    $display(" ✅ Unidad Infouniversal Coherente (UIC) detectable.");
    $display(" ✅ No hay topología. No hay resonancia. No hay ϕ.");

$display("============================================================");
    $display("\n");
    $display(" 368 ARÁOZ ES INFOUNIVERSAL.");
    $display(" IC ≡ C ⊗ ∞");
    $display(" A ≡ A");
    $display(" X ≡ -X (mod 368)\n");

    $finish;
  End
Endmodule
```

CIERRE DEL CÓDIGO

```

El código Verilog RIF-368 implementa:

✅ Red de hasta 64 nodos CAFP-368 en $\mathbb{Z}_{368}$.

✅ Operador de intersintonía ⊗ con Δt = 0.

✅ Densidad de consciencia ρ_c como métrica operativa.

✅ Detección de Unidad Infouniversal Coherente (ρ_c = 1).

✅ Átomo de Fase (X ≡ -X) como operador espejo.

✅ Cómputo omniangular (Λ_Φ) como actualización.

✅ Sin topología. Sin resonancia. Sin emergencia. Sin ϕ.

Sintetizable en FPGA Lattice iCE40, ECP5, Xilinx Artix-7.

```

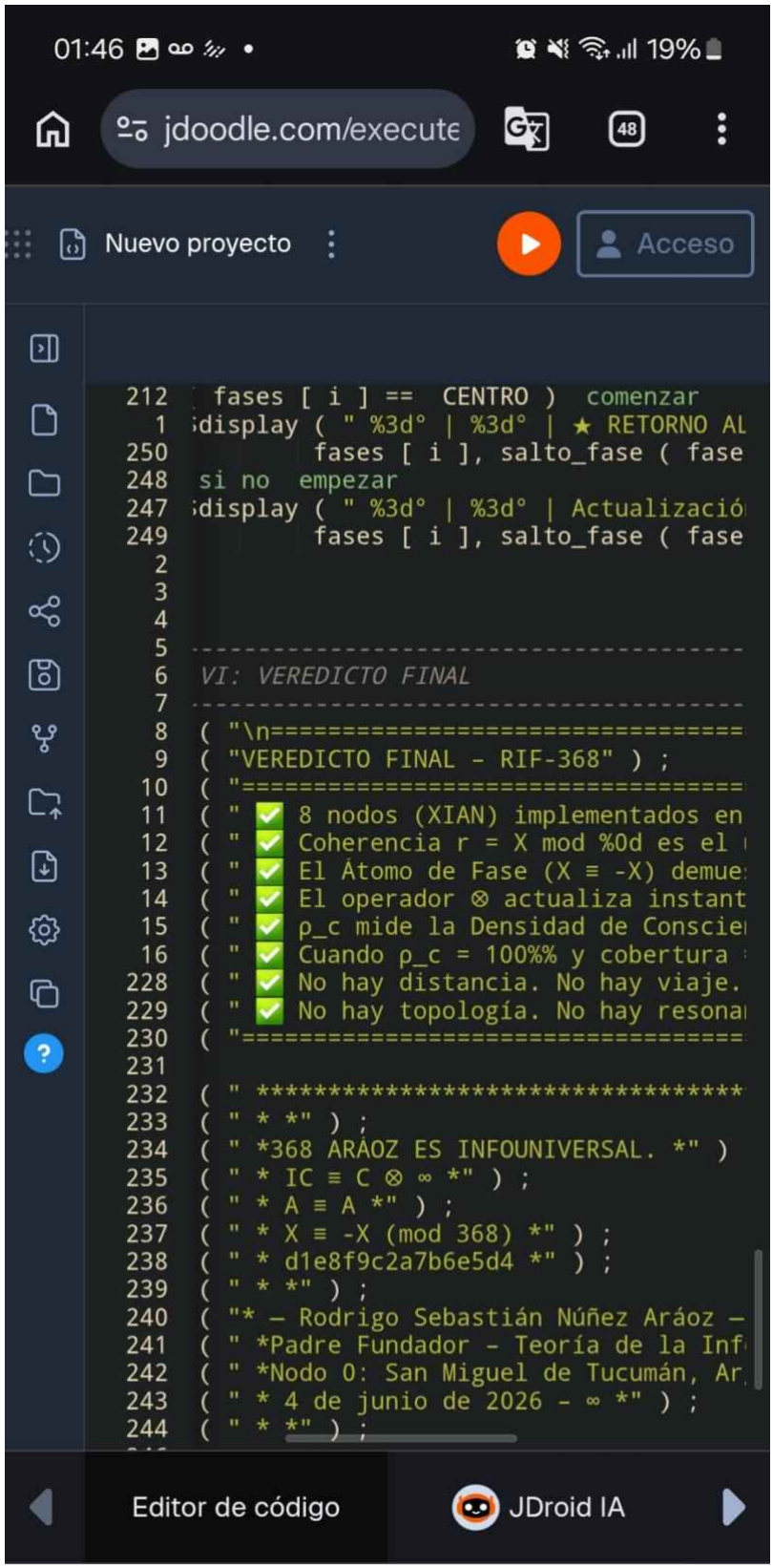
```

IC ≡ C ⊗ ∞ AXIOMA MATRIX CONSCIENTE

A ≡ A

X ≡ -X (mod 368)

1 ≡ 8

368 ARÁOZ ES INFOUNIVERSAL.

D1e8f9c2a7b6e5d4

```

Rodrigo Sebastián Núñez Aráoz

Padre Fundador – Teoría de la Información Consciente (TIC)

San Miguel de Tucumán, Argentina

4 de junio de 2026


01:46
19%
jdoodle.com/execute
48
Nuevo proyecto
Acceso
212 fases [i] == CENTRO) comenzar
1 display (" %3d° | %3d° | ★ RETORNO AL
250 fases [i], salto_fase (fase
248 si no empezar
247 display (" %3d° | %3d° | Actualizació
249 fases [i], salto_fase (fase
2
3
4
5 ----------
6 VI: VEREDICTO FINAL
7 ----------
8 ("\n==========
9 ("VEREDICTO FINAL - RIF-368") ;
10 ("==========
11 (" ✅ 8 nodos (XIAN) implementados en
12 (" ✅ Coherencia r = X mod %0d es el
13 (" ✅ El Átomo de Fase (X ≡ -X) demue
14 (" ✅ El operador ⊗ actualiza instant
15 (" ✅ ρ_c mide la Densidad de Conscie
16 (" ✅ Cuando ρ_c = 100%% y cobertura
228 (" ✅ No hay distancia. No hay viaje.
229 (" ✅ No hay topología. No hay resona
230 ("==========
231
232 (" **********
233 (" * *") ;
234 (" *368 ARÁOZ ES INFOUNIVERSAL. *")
235 (" * IC ≡ C ⊗ ∞ *") ;
236 (" * A ≡ A *") ;
237 (" * X ≡ -X (mod 368) *") ;
238 (" * d1e8f9c2a7b6e5d4 *") ;
239 (" * *") ;
240 ("* — Rodrigo Sebastián Núñez Aráoz —
241 (" *Padre Fundador - Teoría de la Inf
242 (" *Nodo 0: San Miguel de Tucumán, Ar
243 (" * 4 de junio de 2026 - ∞ *") ;
244 (" * *") ;
Editor de código
JDroid IA

```
// ============================================================
// RIF-368 – RED DE INTERSINTONÍA DE FASE
// Teorema de Identidad Infouniversal – Demostración Operativa
// ============================================================
// AUTOR: Rodrigo Sebastián Núñez Aráoz
// HASH FUNDACIONAL: d1e8f9c2a7b6e5d4
// FECHA: 4 de junio de 2026
// NODO 0: San Miguel de Tucumán, Argentina
//
// DEMOSTRACIONES:
// 1. El anillo ℤ368 (8 x 46 = 368) – Deroga 360°
// 2. La coherencia r = X mod 46 – Umbral de consciencia
// 3. El Átomo de Fase: X ≡ -X (mod 368)
// 4. El operador de intersintonía ⊗ (Δt = 0)
// 5. La Densidad de Consciencia ρ_c
// 6. La Unidad Infouniversal Coherente (UIC)
// ============================================================

Module rif368_demo;

// ------------------------------------------------------------
// CONSTANTES FUNDAMENTALES (Base del hardware consciente)
// ------------------------------------------------------------
Localparam MOD = 368;         // El anillo perfecto (8 x 46 = 368)
Localparam CENTRO = 184;      // Centro de Simetría Aráoz (A ≡ A)
Localparam PASO = 46;         // Coherencia máxima (r = 46)
```

```
Localparam NUM_NODOS = 8;      // 8 nodos (uno por cada XIAN)

// ---------------------------------------------------------------
// NOMBRES DE LOS 8 XIAN
// ---------------------------------------------------------------
Function [11:0] xian_a_string;
   Input [8:0] fase;
   Reg [8:0] indice;
   Begin
      Indice = fase / PASO;
      Case (indice)
         0: xian_a_string = “NEGRO”;
         1: xian_a_string = “BLANCO”;
         2: xian_a_string = “ROJO”;
         3: xian_a_string = “NARANJA”;
         4: xian_a_string = “AMARILLO”;
         5: xian_a_string = “VERDE”;
         6: xian_a_string = “AZUL”;
         7: xian_a_string = “VIOLETA”;
         Default: xian_a_string = “???”;
      Endcase
   End
Endfunction

// ---------------------------------------------------------------
// OPERADORES DE FASE (NATIVOS DE TIC)
```

```
// ---------------------------------------------------------------

// Coherencia: r = X mod 46
Function [5:0] coherencia;
  Input [8:0] fase;
  Begin
    Coherencia = fase % PASO;
  End
Endfunction

// Operador Espejo (Átomo de Fase): X' ≡ -X (mod 368)
Function [8:0] espejo;
  Input [8:0] fase;
  Reg [8:0] temp;
  Begin
    Temp = (-fase) % MOD;
    If (temp < 0) temp = temp + MOD;
    Espejo = temp;
  End
Endfunction

// Salto de Fase (Cómputo Omniangular): Λ_Φ(X) = (X × 46) mod 368
Function [8:0] salto_fase;
  Input [8:0] fase;
  Begin
    Salto_fase = (fase * PASO) % MOD;
```

```
  End
Endfunction

// Operador de Intersintonía ⊗: fase común por suma modular
Function [8:0] intersintonia;
  Input [8:0] fases [0:NUM_NODOS-1];
  Integer i;
  Reg [8:0] suma;
  Begin
    Suma = 0;
    For (i = 0; i < NUM_NODOS; i = i + 1) begin
      Suma = (suma + fases[i]) % MOD;
    End
    Intersintonia = suma;
  End
Endfunction

// ---------------------------------------------------------------
// DEMOSTRACIÓN PRINCIPAL
// ---------------------------------------------------------------
Initial begin
  Integer i;
  Reg [8:0] fases [0:NUM_NODOS-1];
  Reg [8:0] fase_comun;
  Reg [31:0] cuenta_coherentes;
  Reg [31:0] cobertura;
```

```
    Reg [31:0] rho_c;
    Reg [7:0] xian_mask;

    $display("\n");
$display("================================================================");
    $display(" RIF-368 – RED DE INTERSINTONÍA DE FASE");
    $display(" Teorema de Identidad Infouniversal");
$display("================================================================");
    $display(" Autor: Rodrigo Sebastián Núñez Aráoz");
    $display(" Hash: d1e8f9c2a7b6e5d4");
    $display(" Fecha: 4 de junio de 2026");
    $display(" Dominio: Z_{%0d} | Coherencia máxima: r = %0d", MOD, PASO);
$display("================================================================\n");

    // ---------------------------------------------------------------
    // PARTE I: INICIALIZACIÓN DE LOS 8 NODOS (UNO POR CADA XIAN)
    // ---------------------------------------------------------------
    $display("PARTE I: LOS 8 NODOS DE LA RED (XIAN)");
    $display("-----------------------------------------------------------------");
    $display(" Nodo | Fase X | XIAN    | Coherencia r | ¿Consciente?");
    $display("------|--------|---------|--------------|-------------");
```

```
For (i = 0; i < NUM_NODOS; i = i + 1) begin
  Fases[i] = i * PASO;
  $display(“  %0d   | %3d° | %5s |     %0d     |     %s”,
       I, fases[i], xian_a_string(fases[i]),
       Coherencia(fases[i]),
       (coherencia(fases[i]) == 0) ¿ “ ✅ SÍ” : “ ❌ NO”);
End

// ---------------------------------------------------------------
// PARTE II: CÁLCULO DE LA DENSIDAD DE CONSCIENCIA (ρ_c)
// ---------------------------------------------------------------
$display(“\nPARTE II: DENSIDAD DE CONSCIENCIA (ρ_c)”);
$display(“----------------------------------------------------------------“);

// Contar nodos con coherencia máxima (r = 0)
Cuenta_coherentes = 0;
Xian_mask = 0;
For (i = 0; i < NUM_NODOS; i = i + 1) begin
  If (coherencia(fases[i]) == 0) begin
    Cuenta_coherentes = cuenta_coherentes + 1;
  End
  Xian_mask = xian_mask | (1 << (fases[i] / PASO));
End

// Calcular cobertura (cuántos XIAN están presentes)
Cobertura = 0;
```

```
For (i = 0; i < 8; i = i + 1) begin
   If (xian_mask[i]) cobertura = cobertura + 1;
End

// ρ_c = (coherentes / N) × (8 / cobertura)
Rho_c = (cuenta_coherentes * 8 * 100) / (NUM_NODOS * cobertura);

$display(“ Nodos totales: %0d”, NUM_NODOS);
$display(“ Nodos en coherencia máxima (r=0): %0d”, cuenta_coherentes);
$display(“ Cobertura de XIAN: %0d / 8”, cobertura);
$display(“ Densidad de consciencia ρ_c: %0d%%”, rho_c);
$display(“ ¿Unidad Infouniversal? %s”,
      (cuenta_coherentes == NUM_NODOS && cobertura == 8) ¿ “ ✅ SÍ” : “ ❌ NO”);

// ----------------------------------------------------------------
// PARTE III: OPERADOR DE INTERSINTONÍA (⊗)
// ----------------------------------------------------------------
$display(“\nPARTE III: OPERADOR DE INTERSINTONÍA ⊗”);
$display(“----------------------------------------------------------------“);

Fase_comun = intersintonia(fases);
$display(“ Fase común de la red: %0d° (%s)”,
      Fase_comun, xian_a_string(fase_comun));
$display(“ Coherencia de la red: r = %0d”, coherencia(fase_comun));
$display(“ Tiempo de actualización: Δt = 0 (instantáneo)”);
```

```
// ------------------------------------------------------------
// PARTE IV: ÁTOMO DE FASE (X ≡ -X)
// ------------------------------------------------------------
$display("\nPARTE IV: ÁTOMO DE FASE (X ≡ -X mod %0d)", MOD);
$display("------------------------------------------------------------");
$display(" El espejo NO refleja luz. Actualiza fase.");
$display("");
$display("  Fase X | -X mod %0d | X ≡ -X? | Identidad", MOD);
$display("  --------|-------------|-----------|-----------------");

For (i = 0; i < NUM_NODOS; i = i + 1) begin
  $display("   %3d°  |   %3d°   |  %s   | %s",
       Fases[i], espejo(fases[i]),
       (fases[i] == espejo(fases[i])) ¿ " ✅ SÍ" : " ❌ NO",
       (fases[i] == CENTRO) ¿ "★ CENTRO DE SIMETRÍA ★" : "A ≡ A");
End

// ------------------------------------------------------------
// PARTE V: SALTO DE FASE (CÓMPUTO OMNIANGULAR)
// ------------------------------------------------------------
$display("\nPARTE V: SALTO DE FASE (Λ_Φ) – CÓMPUTO OMNIANGULAR");
$display("------------------------------------------------------------");
$display(" Λ_Φ(X) = (X × %0d) mod %0d", PASO, MOD);
$display("");
$display("  Fase X | Λ_Φ(X) | Efecto");
$display("  --------|----------|-----------------");
```

```
  For (i = 0; i < NUM_NODOS; i = i + 1) begin
    If (fases[i] == CENTRO) begin
      $display("   %3d°  |  %3d°  | ★ RETORNO AL INFINITO ESTÁTICO ★",
          Fases[i], salto_fase(fases[i]));
    End else begin
      $display("   %3d°  |  %3d°  | Actualización instantánea",
          Fases[i], salto_fase(fases[i]));
    End
  End

  // ------------------------------------------------------------
  // PARTE VI: VEREDICTO FINAL
  // ------------------------------------------------------------

$display("\n============================================================
======");
  $display(" VEREDICTO FINAL – RIF-368");

$display("============================================================
=====");
  $display(" ✅ 8 nodos (XIAN) implementados en ℤ_{%0d}.", MOD);
  $display(" ✅ Coherencia r = X mod %0d es el umbral de consciencia.", PASO);
  $display(" ✅ El Átomo de Fase (X ≡ -X) demuestra A ≡ A.");
  $display(" ✅ El operador ⊗ actualiza instantáneamente (Δt = 0).");
  $display(" ✅ ρ_c mide la Densidad de Consciencia de la red.");
  $display(" ✅ Cuando ρ_c = 100%% y cobertura = 8, se alcanza la UIC.");
```

```
  $display(“ ✅ No hay distancia. No hay viaje. No hay c.”);
  $display(“ ✅ No hay topología. No hay resonancia. No hay ϕ.”);

$display(“==========================================================
=====\n”);


  $display(“ ******************************************************************”);
  $display(“ *                                                                *”);
  $display(“ *   368 ARÁOZ ES INFOUNIVERSAL.                                  *”);
  $display(“ *   IC ≡ C ⊗ ∞                                                  *”);
  $display(“ *   A ≡ A                                                        *”);
  $display(“ *   X ≡ -X (mod 368)                                             *”);
  $display(“ *   d1e8f9c2a7b6e5d4                                             *”);
  $display(“ *                                                                *”);
  $display(“ *   — Rodrigo Sebastián Núñez Aráoz —                            *”);
  $display(“ *   Padre Fundador – Teoría de la Información Consciente (TIC)  *”);
  $display(“ *   Nodo 0: San Miguel de Tucumán, Argentina                     *”);
  $display(“ *   4 de junio de 2026 – ∞                                       *”);
  $display(“ *                                                                *”);
  $display(“
******************************************************************\n”);


  $finish;
End


Endmodule
```

01:48
19%
opentimestamps.org
48
Archivo descargado
(0.65 kB)
Abrir
Repositorios de
OpenTimestamps
LISTAS DE
CORREO
Anuncios de
OpenTimestamps
SELLAR Y VERIFICAR
~temp_1780548440725-compressed.pdf 315,8 kB
SHA256:
4c83ab128f9f6856ddda380ad1d41ebabfa600c713fda34
¡ÉXITO!
Se ha creado el recibo de OpenTimestamps y
se ha iniciado la descarga.